

CLAIMS

None of the claims has been amended. The claims are reproduced here for the Examiner's convenience.

1. (Previously Presented) A method in a computer operating system for associating an identifier with a plurality of processes, the method comprising:

- starting a first process;
- associating the first process with the identifier;
- creating, from the first process, a second process;
- associating the second process with the identifier;
- associating a third process with the identifier, wherein the third process is not a descendant of the first process;
- creating, from the third process, a fourth process; and
- associating the fourth process with the identifier.

2. (Previously Presented) The method of claim 1 wherein the plurality of processes together provide the functionality of a dedicated server application program.

3. (Previously Presented) The method of claim 1 wherein the first process comprises a system initialization process.

4. (Previously Presented) The method of claim 1 wherein associating the first process with the identifier comprises storing an entry in a data structure, the entry comprising the identifier and a process identification number of the first process.

5. (Previously Presented) The method of claim 1 further comprising intercepting a system call, made by the first process, that creates a process.

6. (Previously Presented) The method of claim 5 wherein intercepting the system call comprises replacing a first pointer, which points to object code of the system call, with a second pointer, which points to different object code, such that making the system call causes the different object code to execute.

7. (Previously Presented) The method of claim 6 further comprising inserting the different object code into the operating system.

8. (Previously Presented) The method of claim 7 wherein inserting the different object code into the operating system comprises loading a module into a running operating system kernel, the module comprising the different object code.

9. (Previously Presented) The method of claim 1 further comprising:

loading, by a modified loader program, the first process into computer memory;
starting, by the modified loader program, the first process; and
storing, by the modified loader program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of the first process.

10. (Previously Presented) The method of claim 1 further comprising:

starting, by a virtual process manager program, the first process; and

storing, by the manager program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of the first process.

11. (Previously Presented) A computer-readable medium storing a computer program product for using a computer operating system to associate an identifier with a plurality of processes, the computer program product comprising:

- program code for starting a first process;
- program code for associating the first process with the identifier;
- program code for creating, from the first process, a second process;
- program code for associating the second process with the identifier;
- program code for associating a third process with the identifier, wherein the third process is not a descendant of the first process;
- program code for creating, from the third process, a fourth process; and
- program code for associating the fourth process with the identifier.

12. (Previously Presented) The computer-readable medium of claim 11 wherein the computer program product further comprises program code for storing an entry in a data structure, the entry comprising the identifier and a process identification number of the first process.

13. (Previously Presented) The computer-readable medium of claim 11 wherein the computer program product further comprises:

- program code for intercepting a system call that creates a process; and

program code for associating the process being created with the identifier of a process that made the system call.

14. (Previously Presented) The computer-readable medium of claim 11 wherein the computer program product further comprises:

program code comprising instructions to associate the process being created with the identifier of the process that made the system call; and
program code for replacing a pointer to the system call with a pointer to the instructions to associate, such that making the system call causes the instructions to associate to execute.

15. (Previously Presented) The computer-readable medium of claim 14 wherein the computer program product further comprises program code for storing the instructions to associate in the operating system by loading a module into a running operating system kernel, the module comprising the instructions to associate.

16. (Previously Presented) The computer-readable medium of claim 11 wherein the computer program product further comprises:

program code for starting the first process; and
program code for storing an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of the first process.

17. (Previously Presented) The computer-readable medium of claim 16 wherein the computer program product further comprises program code for loading the first process into computer memory.

18. (Previously Presented) A computer operating system for associating an identifier with a plurality of processes, the computer operating system comprising:

an initiation module, for starting a first process and a third process, wherein the

third process is not a descendant of the first process;

a creation module, coupled to the initiation module, for creating, from the first

process, a second process, and for creating, from the third process, a fourth process; and

an association module, coupled to the creation module, for associating the first

process with the identifier, and for associating the second process with the identifier, and for associating the third process with the identifier, and for associating the fourth process with the identifier.

19. (Previously Presented) The system of claim 18 further comprising a storage module, coupled to the association module, for storing an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of the first process.

20. (Previously Presented) The system of claim 18 further comprising an interception module, coupled to the association module, for intercepting a system call that creates a process; and wherein the association module is further for associating the process being created with the identifier of a process that made the system call.

21. (Previously Presented) The system of claim 20 further comprising a storage module, coupled to the interception module, for storing object code comprising instructions to associate the process being created with the identifier of the process that made the system call; and wherein the interception module is further for replacing a pointer to the system call with a pointer to the stored object code, such that making the system call causes the object code to execute.

22. (Previously Presented) The system of claim 21 further comprising an insertion module, coupled to the storage module, for inserting the instructions to associate in the operating system by loading a module into a running operating system kernel.

23. (Previously Presented) A method in a computer operating system for associating an identifier with a plurality of processes, the method comprising:

executing a system initialization process;

associating the system initialization process with the identifier;

associating a second process, created from the system initialization process, with the identifier;

associating a third process with the identifier, wherein the third process is not a descendant of the system initialization process;

creating, from the third process, a fourth process; and

associating the fourth process with the identifier.

24. (Cancelled)

25. (Previously Presented) A method in a computer operating system for associating an identifier with a plurality of processes, the method comprising:

loading, by a modified loader program, a first process into computer memory; and
storing, by the modified loader program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of the loaded process;

storing, by the modified loader program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of a second process, wherein the second process was created from the first process;

storing, by the modified loader program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of a third process, wherein the third process is not a descendant of the first process;

storing, by the modified loader program, an entry in a data structure in computer memory, the entry comprising the identifier and a process identification number of a fourth process, wherein the fourth process was created from the third process.

26. (Previously Presented) A method in a computer operating system for associating an identifier with a plurality of processes, the method comprising:

associating a first process with the identifier;

associating a second process with the identifier, wherein the second process is not a descendant of the first process;

creating, from the second process, a third process; and
associating the third process with the identifier.